

Novel Design and Evaluation of Redirection Controllers using Optimized Alignment and Artificial Potential Field

Xue-Liang Wu, Huan-Chang Hung , Sabarish V. Babu , and Jung-Hong Chuang 

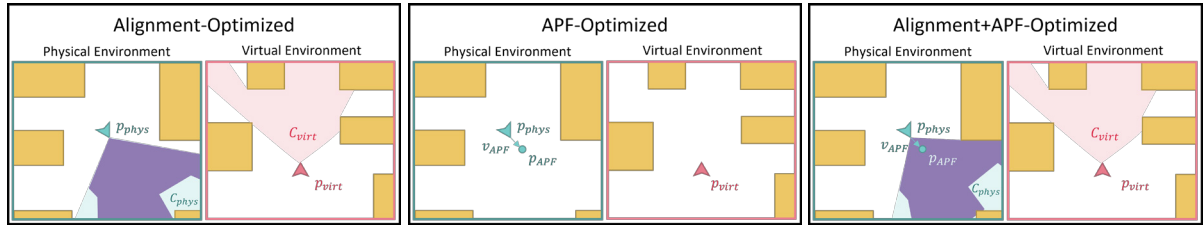


Fig. 1: The novel proposed RDW controllers: Alignment-Optimized controller (left), APF-Optimized controller (middle), and Alignment+APF-Optimized controller (right). Alignment-Optimized controller: For the user's position in VE and PE, C_{virt} (in pale pink) and C_{phys} (in light blue) are the virtual cell and the corresponding physical cell, respectively, representing the walkable areas in front of the user in both environments. We superimpose C_{virt} onto C_{phys} , and the area in purple is the overlapped area of the two cells. Let $C_{phys}(\theta)$ be the physical cell with its bisecting vector deviated from the user's heading by an angle θ . The objective function for the Alignment-Optimized controller considers the relative area of the overlapped area $Area(C_{virt}(\theta) \cap C_{phys}(\theta))$ over the area of the virtual cell $Area(C_{virt}(\theta))$. The optimized θ^* is then used to finely set the RDW gains. APF-Optimized controller: V_{APF} is a vector originating from the user's position p_{phys} , and p_{APF} is a point lying on V_{APF} , located 0.50m away from p_{phys} . The objective function for the APF-Optimized controller considers the finite difference of the directional derivative of APF along $V_{APF}(\theta)$, where θ is the angle between $V_{APF}(\theta)$ and the user's heading. The optimized θ^* represents the direction for the maximum decrease of APF within an angle domain constrained by the minimum and maximum curvature gains, and is used to finely set the RDW gains. Alignment+APF-Optimized controller: By minimizing both objective functions simultaneously, we use the optimized θ^* to finely set the RDW gains.

Abstract— Redirected walking allows users to naturally locomote within virtual environments that are larger than or different in layout from the physically tracked space. In this paper, we proposed novel optimization-driven alignment-based and Artificial Potential Field (APF) redirected walking controllers, as well as an integrated version of the two. The first two controllers employ objective functions of one variable, which is the included angle between the user's heading vector and the target vector originating from the user's physical position. The optimized angle represents the physical cell that is best aligned with the virtual cell or the target vector on which the designated point has the minimum APF value. The derived optimized angle is used to finely set RDW gains. The two objective functions can be optimized simultaneously, leading to an integrated controller that is potentially able to take advantage of the alignment-based controller and APF-based controller. Through extensive simulation-based studies, we found that the proposed alignment-based and integrated controllers significantly outperform the state-of-the-art controllers and the proposed APF based controller in terms of the number of resets. Furthermore, the proposed alignment controller and integrated controller provide a more uniform likelihood distribution across distance between resets, as compared to the other controllers.

Index Terms—Alignment, Artificial Potential Field, Redirected Walking, Virtual Reality

1 INTRODUCTION

Navigating virtual environments (VEs) larger than the available physical environments (PEs) requires many locomotion techniques, such as flying, steering, walking in place, teleportation, and natural walking [49]. Natural walking has been shown to improve the user's sense of presence [49], efficient navigation [37, 45], and spatial knowledge acquisition in VEs [38, 57]. Redirected walking (RDW) is a locomotion technique that supports natural walking by imperceptibly steering the user along a physical path that differs from their virtual path and encounters a minimum number of collisions with physical obstacles [36]. The amount of redirection applied to steer the user is controlled by gains. For the user to remain imperceptible during steering, the gains applied need to be within the perceptual thresholds [42]. The redirection controller determines the gains to steer the user to avoid collision

with obstacles. If an incoming collision is detected, the redirection controller issues a reset to reorient the user away from the obstacles. This reorientation will interrupt the immersive experience, so reducing the number of resets is an ultimate goal for redirected walking research.

Many redirection controllers have been proposed in the last two decades. These controllers can be reactive or predictive [21, 31]. Reactive controllers steer the user based only on the current user state, i.e., the user's position and heading at the current step. In contrast, predictive controllers steer the user based on predictions of the user's future movement in a VE. Predictive controller can usually outperform reactive controllers since predicted paths in VE are also used to decide gains. However, predictive controllers rely on accurate predictions of the user's future trajectory, which is not always available. Recently, the well-known technique of Artificial Potential Field (APF) in robotics for path planning and collision avoidance [17, 18] has been used to design a reactive APF controller that takes into account the layout of PE and utilizes the APF of PE to set appropriate gains to avoid obstacles [5, 28, 48]. For example, the APF controller based on Steer-to-Gradient (APF-S2G), presented in Thomas et al. [48], steers the user according to the negative gradient of the APF and was shown to significantly outperform the classic Steer-to-Center (S2C) controller [35].

- Xue-Liang Wu, Huan-Chang Hung and Jung-Hong Chuang are with National Yang Ming Chiao Tung University. E-mail: {xlwu.cs09, hchung.cs10}@nycu.edu.tw, jhchuang@cs.nycu.edu.tw.
- Sabarish V. Babu is with Clemson University. E-mail: sbabu@clemson.edu.

Manuscript received 25 March 2023; revised 17 June 2023; accepted 7 July 2023.

Date of current version 31 October 2023.

Digital Object Identifier no. 10.1109/TVCG.2023.3320208

More recently, new redirection controllers based on the concept of aligning walkable areas in front of the user in physical and virtual environments (PE and VE) have been proposed [52, 53], based on the observation that if the user's walkable areas in PE and VE are perfectly aligned, they will not encounter a collision with PE while moving inside the walkable area in VE. These are reactive controllers but they utilize information from both PE and VE to steer the user into the aligned walkable area in PE. ARC presented in [52] was the first alignment-based controller reported to significantly outperform APF-S2G and S2C [48]. The visibility-polygon-based (Vis.-Poly.) alignment controller proposed in [53] measures the alignment of walkable areas based on the decomposition of visibility polygons for the user in PE and VE, and was claimed to significantly outperform ARC, APF-S2G, and S2C. Alignment-based redirected walking, especially visibility polygon-based (Vis.-Poly.) alignment, is very promising since it utilizes information from both the PE and VE. However, it has been stated that the visibility polygon-based alignment can be further improved by using visibility polygons to their full potential [53]. The performance of the Vis.-Poly. controller strongly depends on the triangulation of the visibility polygon, but the triangulation can be affected by the layout of obstacles, in which small and distant obstacles can greatly change the triangulation, even though they are unlikely to collide with the user. The APF or Vis.-Poly. controllers search for an appropriate target direction for setting the curvature gain; thus, they are, in nature, an optimization problem. However, instead of optimization computation, the APF-S2G controller uses the negative gradient of APF as the optimal steering direction, and Vis.-Poly. searches for the optimal steering direction among the decomposed slices of the visibility polygon.

Major Contributions: We consider searching for the optimal steering direction in the alignment controller to be an optimization problem and have designed a framework with which not only more accurate alignment measures can be easily derived, but also represented as objective functions, leading to an optimization-driven framework for redirected walking. The objective function involves one variable, which is the included angle between the user's heading and the target direction, both originating from the user's physical position. The optimized angle found represents the best-matched walkable physical area for the given virtual walkable area, and is then used to finely set RDW gains. With the purpose of considering APF while optimizing the alignment measure, we have designed an objective function that represents the local APF value in front of the user's physical position. The optimized angle represents the direction of the local maximum decrease of APF in front of the user. The two proposed objective functions can be optimized simultaneously, leading to a third controller that is potentially able to take advantage of both the alignment-based controller and the APF-based controller. Finally, we present an evaluation showing that the proposed alignment controller and integrated controller outperform Vis.-Poly., ARC, APF-S2G, and APF-RDW [28], and the proposed integrated controller and alignment-based controller significantly outperform the proposed APF-based controller. Moreover, the proposed alignment controller and integrated controller provide a more uniform and consistent likelihood distribution across distance between resets, as compared to the other controllers.

The main contributions of our work include: (a) *An optimization-driven framework for RDW controllers.* (b) *Three novel redirected walking controllers based on the proposed optimization framework namely, an Alignment controller based on a new alignment definition, an APF controller that uses the search for local optimal, and controller that integrates the proposed alignment and APF controllers.* (c) *Finally, we compare and contrast the proposed controllers to the best existing state-of-the-art controllers with regards to number of resets and the likelihood distribution of the reset distances.*

2 RELATED WORK

One way to support the exploration of VEs with natural walking within a smaller physical space is to visualize virtual counterparts of physical obstacles to encourage natural collision avoidance in PE, leading to the concept of Substitutional Reality (SR) [39]. The concept of SR has been adopted to create virtual spaces [15, 23, 24, 40, 41, 50]. Further

research has been conducted to offer different ways of generating virtual counterparts of physical obstacles [7, 26]. However, the distance one can travel with SR in VE is constrained by the physical space. Kwon et al. [19] addressed this issue by combining SR with the resetting technique. Alternatively, Redirected Walking (RDW) proposed by Razzaque et al. [36] enables users to explore VEs with natural walking within a limited physical space by slowly rotating or translating the VE around the user while they walk. The magnitude of the rotation or translation of the VE is controlled by "gains". Three main gains defined by Steinicke et al. [43] are translation, rotation, and curvature gains. Translation gains scale the user's forward steps by translating the VE forward or backward as the user walks in a straight line in the VE. Rotation gains are similar to translation gains, but the scaling is applied to the user's orientation. Curvature gains rotate the VE around the user while they are walking in a straight virtual line, steering the user on a curved physical path. The result of applying redirection gains should be imperceptible to the user. Steinicke et al. [42] reported that the thresholds of translation gain are from 0.86 to 1.26, the thresholds of rotation gain are from 0.67 to 1.24, and curvature gains need to be greater than a 22.03 meter radius. However, the maximum curvature gain with 7.5 meter radius reported in [13] is commonly used in recent literature [47, 48, 52, 53].

The steering of the user's movement in physical space is governed by the redirection controller, which consists of a steering component and a resetting component, with the aim of minimizing the number of collisions the user has with obstacles. The steering component applies redirection gains to guide the user along a physical path, while the resetting component is responsible for detecting incoming collisions and reorienting the user away from the obstacles. Existing resetting techniques are either in-place [5, 8, 48, 51, 55], where the user is re-oriented but not repositioned, or out-of-place [51, 58], where the user is guided to move to a safe location. While the resetting is normally triggered when the user is about to collide with physical obstacles, Xu et al. [56] proposed manipulating the resetting position with no increase in reset number, while avoiding interference to the user.

Redirection controllers can be classified as reactive, predictive, or scripted [21, 31]. Reactive controllers steer the user's movement based on the information available at the current frame, including the user's position and heading in the physical and virtual environments. Most existing controllers are reactive, including classic Steer-to-Center (S2C) [35], Steer-to-Orbit (S2O) [35], and Steer-to-Multiple Targets [35], as well as controllers based on reinforcement learning [20, 44] and artificial potential fields (APF) [5, 48]. These controllers do not require information on the VE, making them more flexible but less optimized in performance. More recently, alignment-based controllers have utilized information on both PE and VE, aiming to steer the user to an area that is compatible with the walkable area on VE [52, 53]. In addition to the information available on the current frame, predictive controllers also utilize the information of the user's predicted virtual path [30, 59]. The performance of predictive controllers is highly dependent on the accuracy of the predicted future paths in VEs. It is, however, difficult and computationally expensive to precisely extract the user's future path. Scripted controllers steer the user as they travel along virtual paths pre-determined by the developers [36]. Since the virtual paths and the redirection technique to be used are predefined by the developers, scripted controllers are effective at incurring fewer collisions but cannot be easily applied to other VEs. While many redirection controllers have been proposed, there have been open-source frameworks proposed, such as The Redirected Walking Toolkit [1] and OpenRDW [22].

After 2018, several reactive controllers based on Artificial Potential Field (APF) [17, 18] were proposed. All of them utilized the "repulsive force" to "push" the user away from the obstacles. Bachmann et al. [5] proposed an APF-based controller, referred to as APF-RDW, to handle scenarios involving multiple users. APF-RDW sums up the repulsive force exerted from each obstacle's or user's boundary to a point and uses the resulting vector at the point to steer the user. Simulation-based and live studies have shown that APF-RDW can outperform S2C in a rectangular empty physical space. Messinger et al. [28] proposed a

refined APF-RDW, called APF-SC, to account for irregular physical space. A simulation study revealed that APF-SC outperformed S2C in terms of the number of resets and the space utilization for empty space with irregular boundaries. Dong et al. [8] added "attractive force" into APF-SC to "pull" the user to a target position which is the farthest position from the other users. Dong et al. [9] proposed the concept of dynamic density-based redirected walking that considers the density of users and boundary to steer the users. Thomas et al. proposed Push/Pull Reactive (P2R) algorithm that uses repulsive force to push the user away from obstacles and attractive force to pull the user towards desire goal [48], with aim to support non-convex physical spaces with interior obstacles. For multi-user RDW, space subdivision methods subdivide a shared PE and allocate a sub-space to each user. Jeon et al. [14] proposed a dynamic space partitioning method utilizing deep reinforcement learning to obtain the optimal partition strategy.

The recently proposed alignment-based controllers are reactive, but they also utilize information from VEs [52, 53]. Different from controllers such as S2C and APF-S2G, which steered the user away from physical obstacles, alignment-based controllers steer the user into a physical walkable area that best matches or aligns with the virtual walkable area. The rationale behind this is that if the user's walkable areas in the PE and VE are perfectly aligned, they will not encounter a collision with the PE while moving inside the walkable area in the VE. The walkable area is constrained inside the visibility polygon of the user, which is the space that is visible to them from their position. The alignment-based controllers measure the similarity of physical and virtual walkable areas based on features such as shape or size, and search for a physical walkable area that most closely matches or is aligned with the virtual walkable area as the steering direction. The ARC presented in [52] steers the user such that their distances to obstacles in the user's heading, left, and right directions in PE and VE match as closely as possible. ARC was reported to outperform the classic S2C and APF-S2G controllers [48]. The visibility-polygon based controller proposed in [53] steers the user to regions of the visibility polygon in the physical environment that match, as closely as possible, the region the user is heading to in the visibility polygon of the VE. The visibility-polygon based controller was reported to outperform ARC, APF-S2G, and S2C controllers [53]. Visibility-polygon-based controllers are currently the best RDW controllers. However, as noted by Williams et al. [53], they can be further improved by utilizing visibility polygons to their full potential. Potential issues include how to decompose the visibility polygon, how to decide on virtual slices, and how to measure the shape similarity between slices of the physical and virtual visibility polygons. In the relation to alignment, the ENI metric proposed by Williams et al. [54] aims to measure the similarity between PE and VE for natural walking in VR.

3 REDIRECTED WALKING USING OPTIMIZED ALIGNMENT AND ARTIFICIAL POTENTIAL FIELD

3.1 Objective Functions for Alignment Controller

Our goal in this paper is to design a framework with which more accurate alignment measures can be easily derived and represented as objective functions, leading to an optimization-driven framework for RDW controllers. For a user's location and heading in a VE, we define a virtual cell that is a 2D view volume-like polygon bounded by the visibility polygon and two rays originating from the user's location, symmetric to the user's heading and forming an included angle of a prescribed degree. We also define a physical cell in a PE in a similar manner. We then superimpose the virtual cell onto the PE such that it is coincident with the physical cell at the endpoint and the two edges adjacent to the endpoint. The superposition and coincidence of physical and virtual cells makes it easier to measure their shape similarity based on area and, more importantly, makes new definitions of alignment possible.

Visibility Polygons Given a position p and heading V in an environment E , the visibility polygon for p with the environment E is the unbounded polygonal region of all points visible from p . We compute the visibility polygon using the algorithm proposed by Suri

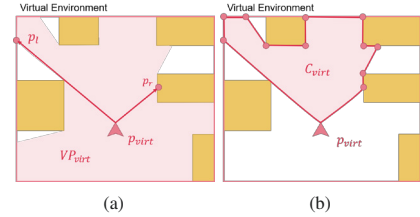


Fig. 2: (a) Cast two rays that are symmetric with respect to the user's heading with an included angle. (b) Check which vertices of VP_{virt} (in pale pink) fall within the area bounded by the two rays.

et al. [46]. The visibility polygon VP is defined by p and a set of n vertices $\{v_0, v_1, \dots, v_{n-1}\}$. Each pair of consecutive vertices v_i and v_{i+1} form an edge of VP . Assume that the user is located in the VE at p_{virt} and with a heading of V_{virt} , and is positioned in the PE at p_{phys} and with a heading of V_{phys} . We first compute the visibility polygons VP_{phys} and VP_{virt} for the user's poses at PE and VE and then derive their virtual cell and physical cell.

Virtual Cell and Physical Cell The virtual cell C_{virt} for the user's position p_{virt} and heading V_{virt} is defined as a 2D view-volume-like polygon originating from p_{virt} with V_{virt} as the viewing direction and a prescribed included angle, bounded by VP_{virt} . The included angle is set to 104° , which is the common field of view (FOV) of the current head mounted display (HMD), such as Oculus Quest2. To derive C_{virt} , we cast two rays that are symmetric with respect to the user's heading with the included angle and compute their intersection points with the visibility polygon VP_{virt} ; see Fig. 2a. Denote the intersection points as p_l and p_r . For each vertex v of VP_{virt} , we check if v falls inside the area bounded by the two rays. Let us denote the set of resulting vertices as S . The virtual cell C_{virt} will be formed by p_{virt} , p_l , vertices in S , and p_r , as seen in Fig. 2b. The physical cell for the user's position p_{phys} and heading V_{phys} in PE can be derived similarly.

Optimization-Driven Alignment Once the virtual cell C_{virt} and the physical cell C_{phys} are ready, we superimpose C_{virt} onto PE such that C_{virt} is coincident with C_{phys} at the user's position and its two adjacent edges. With this superposition, we can define an area-based measure for shape similarity between the physical cell and virtual cell as in [53] by $|Area(C_{phys}) - Area(C_{virt})|$. Let $C_{phys}(\theta)$ denote the physical cell with the same included angle as C_{virt} , and its bisection direction deviating from the user's heading V_{phys} by an angle θ . We can define the shape similarity measure between $C_{phys}(\theta)$ and C_{virt} as the following objective function

$$AlignmentSS(\theta) = |Area(C_{phys}(\theta)) - Area(C_{virt})|. \quad (1)$$

The physical cell that best matches the virtual cell C_{virt} is $C_{phys}(\theta^*)$, where θ^* minimizes $AlignmentSS(\theta)$ over a domain bounded by the minimum and maximum thresholds of the curvature gain. Note that representing the alignment measure as a function of θ allows us to find the optimal alignment by using optimization.

The area-based shape similarity measure $AlignmentSS(\theta)$ cannot, however, distinguish between a favorable case and an unfavorable case when the area differences between physical and virtual cells are the same. For example, in Fig. 4a, the left case shows that the physical cell contains the virtual cell, and the right case shows that the virtual cell contains the physical cell, but their differences in area are the same. We know that the first case is more favorable than the second case, but $AlignmentSS(\theta)$ fails to distinguish these two cases. It may also result in some undesired cases, as shown in Fig. 4b, where the similarity measure is high (i.e., the physical cell and virtual cell have similar areas), but the obstacles are close to the user.

We take a different approach to finding a physical cell $C_{phys}(\theta)$ that is the best match for the virtual cell C_{virt} , based on how much area the physical space can offer to allow for safe walking within the

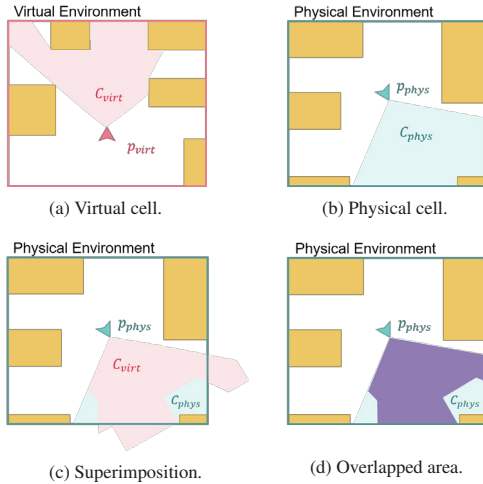


Fig. 3: Process of the alignment measuring.

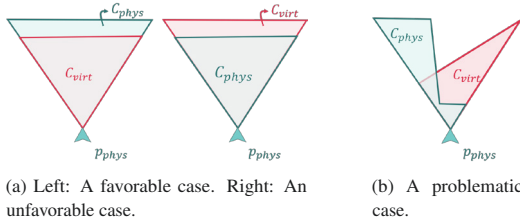


Fig. 4: Cases of area-based shape similarity measure.

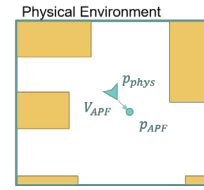
virtual cell C_{virt} . To this end, like $C_{phys}(\theta)$, we denote $C_{virt}(\theta)$ to be the result of rotating C_{virt} by an angle θ from the user's heading V_{phys} in PE. Moreover, we define an alignment measure as the ratio of the overlapped area of $C_{virt}(\theta)$ and $C_{phys}(\theta)$ to the area of the virtual cell C_{virt} , representing the relative area of the virtual cell that is overlapped with the physical cell. The higher the ratio, the higher the alignment. Formally, we define $Alignment(\theta)$ as follows:

$$Alignment(\theta) = 1 - \frac{Area(C_{virt}(\theta) \cap C_{phys}(\theta))}{Area(C_{virt})}, \quad (2)$$

where $Area(C_{virt}(\theta) \cap C_{phys}(\theta))$ represents the area that the physical environment can provide to enable safe walking within the virtual cell C_{virt} . With the formulation of $Alignment(\theta)$, maximizing the ratio is equivalent to minimizing $Alignment(\theta)$. Note that for $Area(C_{virt}(\theta) \cap C_{phys}(\theta))$, we do not need to derive $C_{phys}(\theta)$ and then compute $Area(C_{virt}(\theta) \cap C_{phys}(\theta))$. Instead, $Area(C_{virt}(\theta) \cap C_{phys}(\theta))$ can be replaced by $Area(C_{virt}(\theta) \cap VP_{phys})$.

3.2 Objective Function for APF Controller

While the visibility polygon-based alignment controller performs better than APF-S2G, nevertheless, APF is still a useful source of information for redirected walking. Thus, it is our plan to take the scalar APF function as that of [48] into account when optimizing the proposed objective function $Alignment(\theta)$. To this end, we have designed an objective function for the APF controller, which is also a function of θ . With finite difference, we consider a point p_{APF} lying on the vector $V_{APF}(\theta)$ originating from p_{phys} with an included angle θ from the user's heading V_{phys} . p_{APF} is 0.50m away from the user's position p_{phys} . The APF difference between p_{APF} and p_{phys} divided by the distance between the two points is a finite difference representing the directional derivative of APF along the direction $V_{APF}(\theta)$. We want to find an optimal steering direction θ^* that is the direction with the maximum decrease in directional derivative of APF within the domain $D = [-\frac{1}{7.5}, \frac{1}{7.5}]$, where 7.5 m is the maximum radius for the curvature

Fig. 5: Define $APF(\theta)$ as the APF value at a point on $V_{APF}(\theta)$.

gain [13]. To simplify the finite difference computation, we define the objection function $APF(\theta)$ as the APF value at p_{APF} as follows (See Fig. 5)

$$APF(\theta) = \text{APF value at the endpoint } p_{APF} \text{ of } V_{APF}(\theta). \quad (3)$$

3.3 Optimization-driven Redirected Walking

Suppose that the current configuration is that the user is positioned at p_{phys} with heading V_{phys} in PE, and positioned at p_{virt} with heading V_{virt} in VE. The RDW controller will derive curvature gain, rotation gain, and translation gain and steer the user's movement. In our proposed controllers, the derivation of curvature gain and rotation gain is based on the optimal θ obtained by optimizing $Alignment(\theta)$ or $APF(\theta)$ constrained in the domain $D = [-\frac{1}{7.5}, \frac{1}{7.5}]$, where 7.5 m is the maximal radius for curvature gain [13], so $\frac{1}{7.5}$ is in radian units.

Based on the proposed objective functions, $Alignment(\theta)$ and $APF(\theta)$, we proposed the following three RDW controllers.

- Alignment-Optimized controller:

$$\theta^* = \arg \min_{\theta \in D} Alignment(\theta). \quad (4)$$

- APF-Optimized controller:

$$\theta^* = \arg \min_{\theta \in D} APF(\theta). \quad (5)$$

- Alignment+APF-Optimized controller:

$$\theta^* = \arg \min_{\theta \in D} (Alignment(\theta), APF(\theta)). \quad (6)$$

With Alignment+APF-Optimized controller, we simultaneously optimize $Alignment(\theta)$ and $APF(\theta)$.

Once θ^* is obtained, we compute the curvature gain and rotation gain. Since the search space for the optimization is constrained in $D = [-\frac{1}{7.5}, \frac{1}{7.5}]$, where 7.5 is the maximum radius for curvature gain [13], we set the curvature gain as

$$g_c = \theta^*. \quad (7)$$

Note that the curvature gain is represented in angle per walked meter.

Let r be the signed angle for the user's rotation at p_{virt} . Note that θ^* is also a signed angle. If θ^* and r have the same sign, we apply a maximum rotation gain of 1.24 [42]. A minimum rotation gain of 0.67 [42] is applied when θ^* and r have opposite signs. Specifically, we set the rotation gain g_r as follows:

$$g_r = \begin{cases} 1.24, & \text{if } \theta^* \text{ and } r \text{ have the same sign} \\ 0.67, & \text{if } \theta^* \text{ and } r \text{ have opposite signs.} \end{cases} \quad (8)$$

As depicted in Fig. 6, to compute the translation gain, we first cast a ray originating from p_{phys} with direction θ^* in PE and compute its intersection with the visibility polygon VP_{phys} . Let d_{phys} be the distance from p_{phys} to the intersection point. We also cast a ray originating from p_{virt} with the user's heading direction in VE and compute its intersection with the visibility polygon VP_{virt} . Let d_{virt} be the distance

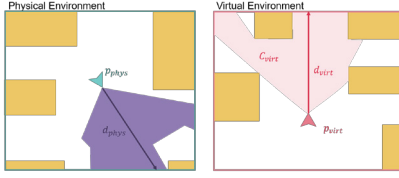


Fig. 6: Cast rays to derive d_{phys} and d_{virt} in PE and VE, respectively.

from p_{virt} to the intersection point. The translation gain g_t is set to be the ratio of d_{phys} and d_{virt} , bounded by thresholds for translation gains.

$$g_t = \text{clamp}\left(\frac{d_{phys}}{d_{virt}}, 0.86, 1.26\right), \quad (9)$$

where 0.86 and 1.26 are the minimum and maximum translation gain [42].

3.3.1 Optimization Solvers

The optimization of $\text{Alignment}(\theta)$ or $\text{APF}(\theta)$ is a constrained optimization with non-differentiable objective functions. We adopted particle swarm optimization (PSO) [16] due to its simplicity of implementation and its capability of finding a global optimum. We carefully set the values for the configurable coefficients in the particle's velocity equation for each iteration step to balance the speed of convergence and accuracy of the solution. We implemented PSO based on the algorithm presented in [25].

In the Alignment+APF-Optimized controller, we aim to optimize both $\text{Alignment}(\theta)$ and $\text{APF}(\theta)$ simultaneously. One simple way to do this is to linearly combine the two objective functions into one. However, before optimizing the resulting objective function, we need to properly normalize both functions, which is not a trivial task for $\text{Alignment}(\theta)$ and $\text{APF}(\theta)$. To avoid the difficulties of normalizing $\text{Alignment}(\theta)$ and $\text{APF}(\theta)$, we instead used the ϵ -constraint method, which is a widely used and more robust method for multi-objective optimization [11]. We implemented the ϵ -constraint method based on the algorithm presented in [11, 27]. With the ϵ -constraint method, we first rank the $\text{Alignment}(\theta)$ and $\text{APF}(\theta)$ based on importance. It has been shown that alignment-based methods, such as ARC and visibility-polygon based methods, significantly outperform the APF-S2G controller [47, 48], so it is reasonable to rank $\text{Alignment}(\theta)$ as more important than $\text{APF}(\theta)$. In this case, an APF-based controller could support the alignment-based controller to prevent the user from getting too close to the obstacles. Using the ϵ -constraint method, we first optimize $\text{Alignment}(\theta)$ constrained in D and obtain the optimal $\theta_{\text{Alignment}}^*$:

$$\theta_{\text{Alignment}}^* = \arg \min_{\theta \in D} \text{Alignment}(\theta). \quad (10)$$

Then we optimize $\text{APF}(\theta)$ as follows:

$$\begin{aligned} \min_{\theta \in D} \quad & \text{APF}(\theta) \\ \text{s.t.} \quad & \text{Alignment}(\theta) \leq (1 + \epsilon) \text{Alignment}(\theta_{\text{Alignment}}^*), \end{aligned} \quad (11)$$

where ϵ represents the amount of tolerance for the optimality of $\theta_{\text{Alignment}}^*$.

3.4 Reset Strategies

Any RDW controller needs resetting strategies to detect when they are about to collide with an obstacle in PE and to reorient the user once it happens. For the trigger condition, we follow the approach proposed in Williams et al. [52]. Once θ^* is computed, two conditions are checked. The first condition is to check if the next position along θ^* is inside a physical obstacle. If so, the reset is triggered; otherwise, the second condition is checked. If the next position is close enough to an obstacle and the angle between the user's heading and the obstacle's normal is greater than 90 degrees, the reset is triggered.

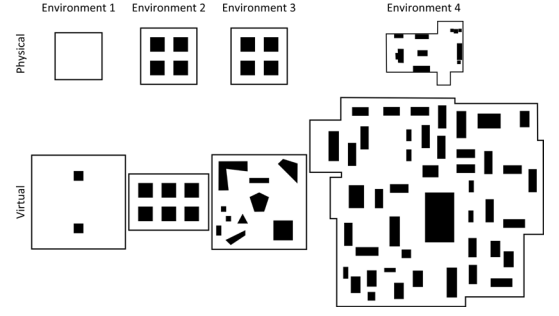


Fig. 7: The layouts of the environment pairs used in our experiments.

Once a reset is triggered, the optimal reorientation angle for the reset will be obtained by optimizing the objective function with θ being the angle between the target direction and the normal n and the constrained domain $[n - 85^\circ, n + 85^\circ]$, where n is the normal of the closest face of the obstacle that triggered the reset. After the optimal reorientation angle is derived, the reorientation process is performed in a way similar to face-center [51] and 2:1-turn [34], with the difference being that the user will end up physically facing the derived optimal direction.

4 EVALUATION DESIGN

We conducted simulation-based experiments to evaluate the performance of the proposed controllers using four pairs of physical and virtual environments. For the experiments, we compared our three controllers against state-of-the-art controllers: APF-RDW presented by Messinger et al. [28], APF-S2G presented by Thomas et al. [48], ARC presented by Williams et al. [52], and Vis.-Poly., the visibility-polygon based alignment controller by Williams et al. [53]. We did not compare our controllers with S2C because it had been used as a baseline for the comparative study in [28, 48, 52, 53]. APF-RDW and APF-S2G were the best redirection controllers when presented in 2019, and are considered to serve as a baseline for this evaluation. ARC is the first alignment-based reactive controller, while Vis.-Poly. controller is currently the best reactive controller.

For each experiment, we simulated the use of redirection controllers for 100 randomly generated paths with a random starting position and orientation, and compared the controllers' performance in terms of the number of resets incurred over the whole simulation and the distance between resets. Both are standard performance metrics for comparative study in redirection walking research. The distance between resets may be redundant since it depends on the number of resets; however, in addition to comparing the average distance between resets, we also wanted to compare their likelihood distributions for different controllers. Before the simulation-based study, we formulated the following hypotheses:

H1 Our Alignment+APF-Optimized controller will perform better than Visibility-Polygon controller [53], ARC [52], APF-RDW [28], and APF-S2G [48] in terms of the number of resets.

H2 Our Alignment-Optimized controller will perform better than Visibility-Polygon controller [53], ARC [52], APF-RDW [28], and APF-S2G [48] in terms of the number of resets.

H3 Our APF-Optimized controller will perform better than APF-RDW [28] and APF-S2G [48] in terms of the number of resets.

H4 Our Alignment+APF-Optimized controller will perform better than the Alignment-Optimized controller in terms of the number of resets.

H5 Our Alignment+APF-Optimized and Alignment-Optimized controller will perform better than the APF-Optimized controller in terms of the number of resets.

H6 Our Alignment+APF-Optimized and Alignment-Optimized controllers will provide a more uniform likelihood distribution across distance between resets as compared to the other redirection controllers.

4.1 Environment Layouts

In designing the test environments, we are aware that the layout or structure of the physical and virtual environments affects the performance of a redirection controller [2, 28]. Moreover, for comparison, we may use the environment pairs that possess sufficient structure complexity and different levels of local similarity and structure complexity, as shown in Fig. 7.

Environment 1 is composed of a $10m \times 10m$ empty PE and a $20m \times 20m$ VE, with two $2m \times 2m$ obstacles placed inside the VE. Note that Environments 2 and 3 come from Williams et al. [53]. Environment 2 has a $12m \times 12m$ PE and a $17m \times 12m$ VE. Both PE and VE have narrow corridors and appear to have a high level of local similarity. Environment 3 consists of a $12m \times 12m$ PE and a $20m \times 20m$ VE, which shares the same PE as Environment 2. The VE has many convex and concave obstacles. We notice that PE and VE have obstacles with different shapes and sizes, thus Environment 3 has lower local similarity than Environment 2. In Environment 4, we used 3D models obtained from the Unity Asset Store and converted them into 2D layout maps. The PE is a $15m \times 12.5m$ living room consisting of sofas, chairs, tables, and cabinets. The VE is a $55m \times 50m$ cityscape map consisting of buildings, which is about 15 times larger than the PE. The local similarity of Environment 4 is considered to be a little lower than Environment 2 because the layouts of PE and VE in Environment 2 are regular and very similar, whereas the layouts of PE and VE in Environment 4 are irregular and different.

4.2 Simulation Design and Settings

Our implementation of APF-S2G is based on the method presented in Thomas et al. [48], APF-RDW is based on [28], ARC is based on [52], and Vis.-Poly. controller is based on [53]. The parameters involved in the simulation were the same as those in [53]. The user was represented as a circle with a radius of $0.5m$, walking with a velocity of $1m/s$, and turning with an angular velocity of $90^\circ/s$. The simulation timestep was $0.05s$. All these four controllers and our controllers share the same condition for detecting the reset, but each controller employs its own method for reorientation. In the implementation of APF-S2G, we used SFR2G for the reorientation in reset. SFR2G takes 10 steps, each in the direction of the negative gradient, and uses the resulting position as the target to reorient the user [48].

For each environment pair, we randomly generated 100 paths. The path lengths were $340 \sim 400m$ for Environment 1, 2 and 3, and $750 \sim 800m$ for Environment 4. The paths used in the simulation studies of [48, 52, 53] consist of a set of waypoints connected by straight lines, and have rotation-in-place at each waypoint. We followed a similar procedure, but used cubic Bézier curves between waypoints. The parameters for generating waypoints were the same as in [2]. To ensure a smooth transition at a waypoint, two adjacent Bézier segments shared the same tangent at the waypoint, which was perpendicular to the bisector of the vectors to the previous and next waypoints. The distance between the waypoint and its adjacent control point was set to be a proportion of the distance between the waypoint and its corresponding neighboring waypoint. The generated piecewise cubic Bézier curve is expected to approximate the global shape of the real user path more closely than the piecewise linear curve, since each Bézier segment is more realistic than a line segment compared to a human locomotion trajectory. However, Bézier segments are smooth while human locomotion trajectories may be more zigzagging. In this case, we can reduce the distance between waypoints. We can also support rotation-in-place at a waypoint by setting different tangent directions for the segments adjacent to the waypoint. We simulated the locomotion on each path for each of the redirection controllers under evaluation. For each path, the user's starting location in the physical and virtual environments and their headings were randomly generated. Note that for each path, all the controllers under evaluation used the same starting locations and headings.

5 RESULTS

5.1 Number of Resets

The Shapiro-Wilk's test showed that some of the data violated the normality assumption, so non-parametric tests were used to analyze the data. Wilcoxon's signed-rank tests were applied for post-hoc comparisons, with p Bonferroni correction, resulting in a significance level of $p < 0.007$ ($0.05/7$). The results of the post-hoc pairwise comparisons are shown in the graphs 8.

5.1.1 Environment 1

The Friedman test revealed a significant difference in the number of resets in the redirection controllers, $\chi^2(6) = 493.761$, $p < 0.001$. The box plot for the number of resets and the results of post hoc Wilcoxon's tests are shown in Fig. 8a. The mean number of resets was 23.41 for Alignment+APF-Optimized ($MED = 24$, $SD = 2.81$), 23.02 for Alignment-Optimized ($MED = 23$, $SD = 2.96$), 26.51 for APF-Optimized ($MED = 27$, $SD = 3.17$), 36.08 for Vis.-Poly. ($MED = 36$, $SD = 3.48$), 44.80 for ARC ($MED = 45$, $SD = 3.85$), 31.41 for APF-RDW ($MED = 32$, $SD = 3.26$), and 26.83 for APF-S2G ($MED = 27$, $SD = 3.04$).

5.1.2 Environment 2

The Friedman test revealed a significant difference in the number of resets among redirection controllers, $\chi^2(6) = 571.724$, $p < 0.001$. The box plot for the number of resets along with the results of post hoc Wilcoxon's tests are depicted in Fig. 8b. The mean number of resets was 84.83 for Alignment+APF-Optimized ($MED = 85$, $SD = 8.84$), 89.52 for Alignment-Optimized ($MED = 89.5$, $SD = 10.22$), 154.04 for APF-Optimized ($MED = 153.5$, $SD = 15.61$), 126.55 for Vis.-Poly. ($MED = 126$, $SD = 9.28$), 146.12 for ARC ($MED = 148$, $SD = 9.70$), 412.10 for APF-RDW ($MED = 419$, $SD = 53.88$), and 477.39 for APF-S2G ($MED = 479$, $SD = 17.77$).

5.1.3 Environment 3

The Friedman test showed a significant difference in the number of resets between redirection controllers, $\chi^2(6) = 567.643$, $p < 0.001$. The box plot for the number of resets along with the results of post-hoc Wilcoxon's tests are shown in Fig. 8c. The mean number of resets was 117.31 for Alignment+APF-Optimized ($MED = 116.5$, $SD = 7.94$), 121.36 for Alignment-Optimized ($MED = 121$, $SD = 10.02$), 171.02 for APF-Optimized ($MED = 172$, $SD = 14.99$), 154.65 for Vis.-Poly. ($MED = 155$, $SD = 11.50$), 172.64 for ARC ($MED = 173$, $SD = 11.09$), 282.92 for APF-RDW ($MED = 281$, $SD = 26.62$), and 364.25 for APF-S2G ($MED = 362$, $SD = 24.45$).

5.1.4 Environment 4

The Friedman test found a significant difference in the number of resets among redirection controllers, $\chi^2(6) = 534.605$, $p < 0.001$. The box plot for the number of resets and the results of post-hoc Wilcoxon's tests are shown in Fig. 8d. The mean number of resets was 108.71 for Alignment+APF-Optimized ($MED = 107$, $SD = 6.67$), 109.11 for Alignment-Optimized ($MED = 108$, $SD = 7.85$), 194.19 for APF-Optimized ($MED = 193.5$, $SD = 23.17$), 145.49 for Vis.-Poly. ($MED = 145$, $SD = 12.69$), 166.12 for ARC ($MED = 164$, $SD = 13.00$), 175.21 for APF-RDW ($MED = 173.5$, $SD = 28.80$), and 773.79 for APF-S2G ($MED = 764.5$, $SD = 209.68$).

5.2 Likelihood of Reset Distance Distribution

We computed the likelihood of the redirection controllers to generate resets in discrete categories of 0.5 meter distances. We computed 41 categories that ranged in reset distances from 0 to $0.5m$ (category 1), $0.5m$ to $1m$ (category 2), $1.0m$ to $1.5m$ (category 3) and so on until greater than $20m$ (category 41). We simulated 100 trials of each controller in each of the test environments, and then computed the distribution of the likelihood of redirection reset distances generated by each controller in that environment. We then computed a multiple regression analysis to evaluate to what extent the redirection controllers and the categories of reset distances predicted the likelihood of generating that reset distance.

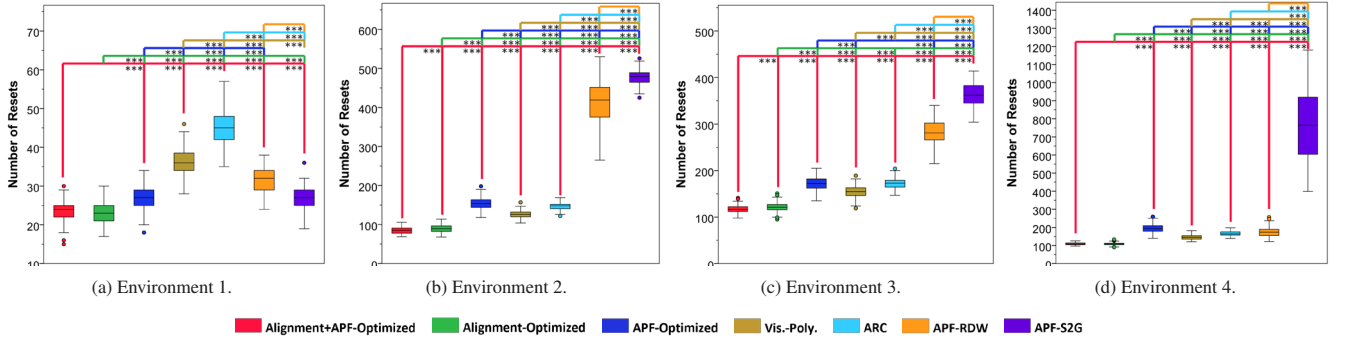


Fig. 8: The box plots of the number of resets for each redirection controller. The colored lines with asterisks (***) indicate the highly significant post-hoc pairwise differences with $p < 0.001$.

Such multiple regression analysis have been employed in VR [4, 10, 29] and similarly in real world studies [6, 32, 33].

The regression analysis contribution will also enable researchers to predict the likelihood of generating a reset distance, given a test environment, redirection controller and reset distance. The curves of the regression profiles also tell us how uniform or lopsided the likelihoods of the resets over reset distance categories are distributed. For instance, if the slope is steep on one end then that informs researchers that the algorithm's frequency of resets in a narrow range of reset distance category, which is mostly in the short distance category, is very high. Thus this is an undesirable characteristic and vice-versa is true - a shallow slope and a uniform curve is more desirable.

We first conducted a multiple regression analysis on the natural log transform of the likelihood distribution across the reset distance categories for each controller, which resulted in a linear relationship between category and likelihoods in each of the 7 controllers. Once we determined through multiple regression that redirection controller type was a significant predictor or affected the log likelihood distribution across reset distances, then we modeled and provided the regression profiles of the likelihood distribution of reset distances for each of the redirection controller in each of the test environments. Redirection controller type was coded in the model as follows: (1) Alignment+APF-Optimized, (2) Alignment-Optimized, (3) APF-Optimized, (4) Visibility-Polygon (Vis.-Poly.), (5) ARC, (6) APF-RDW, (7) APF-S2G. Prior to conducting multiple regression analysis on the log likelihood data, we insured that the histogram of standardized residuals were normally distributed, and the normal P-P plot of the standardized residuals were linear indicating that the data were suitable for linear multiple regression analysis.

5.2.1 Environment 1

In test Environment 1, the multiple regression model on log likelihood of reset distribution data was found to be significant, $F(2, 283) = 8.90$, $p < 0.001$, with an $R^2 = 0.10$. The *log likelihood of reset distances* was equal to $-5.21 + 0.052 \times \text{RedirectionControllerType} + 0.024 \times \text{ResetDistanceCategory}$; log likelihood of reset distance increased by 0.024 for every 0.5m increase in the reset distance category, and a difference of 0.052 was calculated on average between two subsequent redirection types. Both independent variables, *Redirection Controller Type*, $p = 0.045$, and *Reset Distance Category*, $p < 0.001$, were significant predictors of log likelihood of the reset distances. The non-linear regression profiles of how the different redirection controllers affected the distribution of the likelihood of reset distances is shown in Fig. 9a. The cubic regression profile was found to be the best fit for modeling the redirection controller data in the relationship between reset distance category and the likelihood of the reset distance. The cubic regression profiles for the redirection controllers were as follows; Alignment+APF-Optimized: $\text{likelihood} = -0.0086 + 0.0015 \times \text{Category} + 0.000089 \times \text{Category}^2 - 0.0000032 \times \text{Category}^3$ [$R^2 = 0.39$], Alignment-Optimized: $\text{likelihood} = -0.007 + 0.001 \times \text{Category} + 0.00012 \times \text{Category}^2 - 0.0000037 \times$

Category^3 [$R^2 = 0.41$], APF-Optimized: $\text{likelihood} = -0.02 + 0.0044 \times \text{Category} - 0.00005 \times \text{Category}^2 - 0.0000014 \times \text{Category}^3$ [$R^2 = 0.30$], APF-S2G: $\text{likelihood} = -0.02 + 0.0049 \times \text{Category} - 0.000084 \times \text{Category}^2 - 0.00000082 \times \text{Category}^3$ [$R^2 = 0.34$], ARC: $\text{likelihood} = 0.0017 + 0.0057 \times \text{Category} - 0.00024 \times \text{Category}^2 + 0.0000023 \times \text{Category}^3$ [$R^2 = 0.54$], Vis.-Poly.: $\text{likelihood} = -0.0017 + 0.005 \times \text{Category} - 0.00018 \times \text{Category}^2 + 0.0000015 \times \text{Category}^3$ [$R^2 = 0.51$], APF-RDW: $\text{likelihood} = -0.02 + 0.0055 \times \text{Category} - 0.00012 \times \text{Category}^2 - 0.00000042 \times \text{Category}^3$ [$R^2 = 0.33$].

5.2.2 Environment 2

In test Environment 2, the multiple regression model on log likelihood of reset distribution data was found to be significant, $F(2, 234) = 328.70$, $p < 0.001$, with and $R^2 = 0.74$. The *log likelihood of reset distances* was equal to $0.20 - 0.80 \times \text{RedirectionControllerType} - 0.17 \times \text{ResetDistanceCategory}$; log likelihood of reset distance decreased by -0.17 for every 0.5m increase in the reset distance category, and a difference of -0.80 was calculated on average between two subsequent redirection types. Both independent variables, *Redirection Controller Type*, $p < 0.001$, and *Reset Distance Category*, $p < 0.001$, were highly significant predictors of log likelihood of the reset distances. The non-linear regression profiles of how the different redirection controllers affected the distribution of the likelihood of reset distances is shown in Fig. 9b. The quadratic regression profiles for redirection controllers in the relationship between reset distance category and the likelihood of the reset distance were as follows; Alignment+APF-Optimized: $\text{likelihood} = 0.1 - 0.0063 \times \text{Category} + 0.0009 \times \text{Category}^2$ [$R^2 = 0.81$], Alignment-Optimized: $\text{likelihood} = 0.1 - 0.0068 \times \text{Category} + 0.00011 \times \text{Category}^2$ [$R^2 = 0.83$], APF-Optimized: $\text{likelihood} = 0.15 - 0.01 \times \text{Category} + 0.00022 \times \text{Category}^2$ [$R^2 = 0.40$], APF-S2G: $\text{likelihood} = 0.2 - 0.02 \times \text{Category} + 0.00034 \times \text{Category}^2$ [$R^2 = 0.14$], ARC: $\text{likelihood} = 0.14 - 0.01 \times \text{Category} + 0.00019 \times \text{Category}^2$ [$R^2 = 0.64$], Vis.-Poly.: $\text{likelihood} = 0.13 - 0.0094 \times \text{Category} + 0.00016 \times \text{Category}^2$ [$R^2 = 0.68$], APF-RDW: $\text{likelihood} = 0.19 - 0.02 \times \text{Category} + 0.00033 \times \text{Category}^2$ [$R^2 = 0.16$].

5.2.3 Environment 3

In test Environment 3, the multiple regression model on log likelihood of reset distribution data was found to be significant, $F(2, 209) = 656.46$, $p < 0.001$, with and $R^2 = 0.87$. The *log likelihood of reset distances* was equal to $-0.13 - 0.45 \times \text{RedirectionControllerType} - 0.25 \times \text{ResetDistanceCategory}$; log likelihood of reset distance decreased by -0.25 for every 0.5m increase in the reset distance category, and a difference of -0.45 was calculated on average between two subsequent redirection types. Both independent variables, *Redirection Controller Type*, $p < 0.001$, and *Reset Distance Category*, $p < 0.001$, were highly significant predictors of log likelihood of the reset distances. The non-linear regression profiles of how the different redirection controllers affected the distribution of the likelihood of reset distances is

shown in Fig. 9c. The quadratic regression profiles for redirection controllers in the relationship between reset distance category and the likelihood of the reset distance were as follows; Alignment+APF-Optimized: $likelihood = 0.13 - 0.0092 \times Category + 0.00016 \times Category^2$ [$R^2 = 0.74$], Alignment-Optimized: $likelihood = 0.13 - 0.0095 \times Category + 0.00016 \times Category^2$ [$R^2 = 0.76$], APF-Optimized: $likelihood = 0.16 - 0.01 \times Category + 0.00024 \times Category^2$ [$R^2 = 0.40$], APF-S2G: $likelihood = 0.19 - 0.02 \times Category + 0.00033 \times Category^2$ [$R^2 = 0.17$], ARC: $likelihood = 0.15 - 0.01 \times Category + 0.00022 \times Category^2$ [$R^2 = 0.60$], Vis.-Poly.: $likelihood = 0.14 - 0.01 \times Category + 0.00020 \times Category^2$ [$R^2 = 0.67$], APF-RDW: $likelihood = 0.18 - 0.02 \times Category + 0.00030 \times Category^2$ [$R^2 = 0.24$].

5.2.4 Environment 4

In test Environment 4, the multiple regression model on log likelihood of reset distribution data was found to be significant, $F(2, 283) = 392$, $p < 0.001$, with an $R^2 = 0.74$. The log likelihood of reset distances was equal to $-0.68 - 0.39 \times RedirectionControllerType - 0.14 \times ResetDistanceCategory$; log likelihood of reset distance decreased by -0.14 for every 0.5m increase in the reset distance category, and a difference of -0.39 was calculated on average between two subsequent redirection types. Both independent variables, *Redirection Controller Type*, $p < 0.001$, and *Reset Distance Category*, $p < 0.001$, were highly significant predictors of log likelihood of the reset distances. The non-linear regression profiles of how the different redirection controllers affected the distribution of the likelihood of reset distances is shown in Fig. 9d. The cubic regression profile was found to be the best fit for modeling the redirection controller data in the relationship between reset distance category and the likelihood of the reset distance. The cubic regression profiles for the redirection controllers were as follows; Alignment+APF-Optimized: $likelihood = 0.02 + 0.0057 \times Category - 0.00035 \times Category^2 + 0.0000048 \times Category^3$ [$R^2 = 0.79$], Alignment-Optimized: $likelihood = 0.02 + 0.005 \times Category - 0.0003 \times Category^2 + 0.000004 \times Category^3$ [$R^2 = 0.73$], APF-Optimized: $likelihood = 0.12 - 0.01 \times Category + 0.00031 \times Category^2 - 0.000003 \times Category^3$ [$R^2 = 0.47$], APF-S2G: $likelihood = 0.3 - 0.05 \times Category + 0.002 \times Category^2 - 0.00003 \times Category^3$ [$R^2 = 0.24$], ARC: $likelihood = 0.07 - 0.0004 \times Category - 0.00017 \times Category^2 + 0.0000036 \times Category^3$ [$R^2 = 0.86$], Vis.-Poly.: $likelihood = 0.06 + 0.0004 \times Category - 0.00018 \times Category^2 + 0.0000033 \times Category^3$ [$R^2 = 0.87$], APF-RDW: $likelihood = 0.1 - 0.007 \times Category + 0.00018 \times Category^2 - 0.0000016 \times Category^3$ [$R^2 = 0.55$].

5.3 Hypotheses Verification

In terms of the number of resets across all environments, we have the following findings. The Alignment+APF-Optimized and Alignment-Optimized performed significantly better than Vis.-Poly., ARC, APF-RDW, and APF-S2G. Additionally, the Alignment+APF-Optimized and Alignment-Optimized performed significantly better than APF-Optimized. These findings supported **H1**, **H2**, and **H5**. However, the results can only partially support **H3** since the performance of APF-Optimized was not significantly different from APF-S2G in Environment 1, and APF-Optimized performed worse than APF-RDW in Environment 4. The hypothesis **H4** is partially supported since there is no significant difference between Alignment+APF-Optimized and Alignment-Optimized in Environment 1 and Environment 4.

The multiple regression analysis revealed a significant effect of the controllers on the likelihood distribution across different distances between resets. Specifically, the analysis showed that in all environments, the likelihood distributions of the distance between resets were more uniformly distributed across the different categories in the Alignment-APF-Optimized and Alignment-Optimized controllers, whereas for the Vis.-Poly., ARC, APF-RDW and APF-S2G controllers, the likelihood distribution of reset distances across the different categories was found to be uneven. Furthermore, the slopes of the Vis.-Poly., ARC, APF-RDW and APF-S2G controller likelihood distributions revealed a highly non-uniform distribution, such that there was a significantly higher likelihood and frequency of smaller reset distances, and little or

no likelihood and frequency of the larger reset distances. Hence, **H6** is supported.

6 DISCUSSION

The statistical results show that both Alignment+APF-Optimized and Alignment-Optimized controller outperformed the Vis.-Poly. controller in terms of number of resets, as proved by the verification of **H1** and **H2**. This highlights the effectiveness of our proposed alignment definition and optimization-driven framework, which incorporates a more accurate shape similarity measure that considers the overlapping area of the virtual cell and physical cell, along with the fine-grained setting of gains. This improvement leads to better controller performance, demonstrating that the proposed alignment definition and optimization-driven framework with a finer alignment measure and a more precise strategy for setting redirection gains are key factors for improving alignment-based controllers.

The design of the APF-Optimized controller was initially intended to offer a way to incorporate APF information into the optimization process of Alignment-Optimized controller. The study results showed that the APF-Optimized controller performed significantly better than the APF-S2G in Environment 2, 3 and 4. The difference between the two controllers is that APF-S2G takes the negative gradient as the steering direction, while APF-Optimized uses the local minimum directional derivative. We observed that in a narrow long aisle, the negative gradients at points inside the aisle always point to the opposite wall, so in the process of APF-S2G steering, the user is gradually steered towards the wall and may easily end up in a collision. As for the APF-Optimized controller, the optimized direction obtained for steering at points inside the aisle is always within the bounds constrained by the gain thresholds, so the user can be gradually steered towards the center line of the aisle and thus potentially avoid a collision. Moreover, with the optimized direction, we can finely set the gains for RDW. Another possible reason is that the reorientation strategy in the resetting of APF-S2G usually sets the direction pointing to the opposite wall as the steering direction, which can easily result in a collision. However, the results showed that the performance of APF-Optimized controller was not significantly different from APF-S2G in Environment 1. In this environment pair, the steering directions obtained by the APF-Optimized controller were not too dissimilar from those of the APF-S2G, resulting in similar performance. In Environment 4, APF-RDW performed better than the APF-Optimized controller and APF-S2G, because it was often observed that APF-RDW could steer the user to an open space.

Supported by the verification of **H4** in Environments 2 and 3, the incorporation of APF information and alignment measure proved to improve the controller's performance. This demonstrates the potential for achieving fewer resets by considering multiple redirection strategies simultaneously. However, balancing between different strategies is not straightforward, as previously mentioned. The statistical results revealed that our Alignment+APF-Optimized controller achieved varying degrees of improvement compared to the Alignment-Optimized and APF-Optimized controllers in different environment pairs. This highlights that the performance improvement can be affected by the layouts of PE and VE, especially for controllers with hybrid strategies. Consistent with the findings from previous works, alignment-based controllers are superior in performance to APF-based controllers in terms of the number of resets, as proved by the verification of **H5**.

As one of the key improvements we proposed compared to the previous approaches, we expected that the alignment measure which considers the overlapping area of the virtual cell and the physical cell could distinguish between the favorable case, the unfavorable case, and the problematic case, as illustrated in Figures 4a and 4b. Successfully distinguishing between such cases would lead to more uniformly distributed likelihood distributions of reset distances, as the physical cell that offers the most space to allow walking within the virtual cell would be chosen as the steering target to steer the user, which would effectively reduce the number of short-distance resets. As shown by the verification of **H6**, our Alignment+APF-Optimized and Alignment-Optimized controllers did provide more uniform likelihood

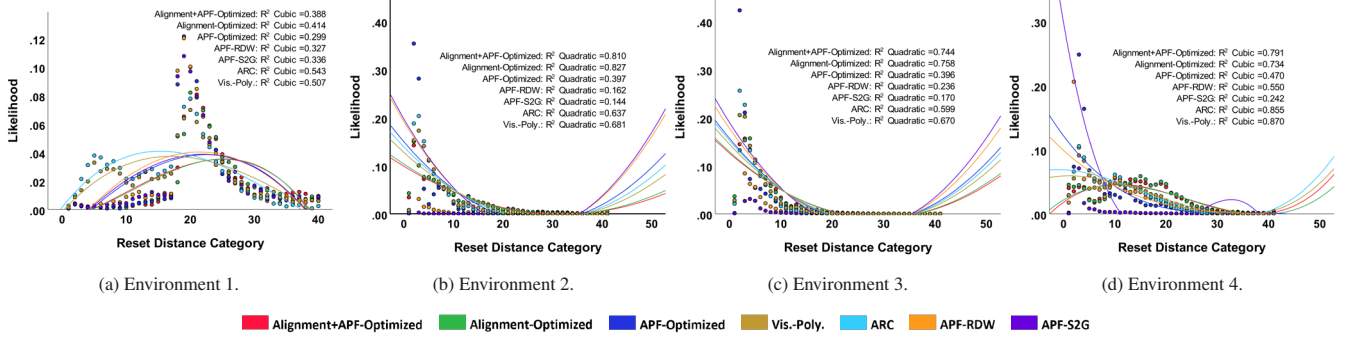


Fig. 9: Cubic regression profiles of the redirection controller's likelihood of reset distance distribution.

distributions across reset distances than other controllers, highlighting the effectiveness of our proposed approach.

The four environment pairs used for the simulation differ in geometric layout and degree of local similarity, which lead to different performance characteristics of each controller across each environment pair. In Environment 1, PE has a square space with no obstacles inside, which is an ideal layout for APF-based controllers [28, 48]. This could be the reason why alignment-based controllers could not perform better than APF-based controllers. In Environment 2, the physical and virtual environment are similar, which is a favorable case for alignment-based controllers due to the relatively low difficulty to achieve collision-free navigation [53]. The results showed that our Alignment-Optimized and Alignment+APF-Optimized controllers could perform significantly better than APF-Optimized, Vis.-Poly., ARC, APF-RDW and APF-S2G. In Environment 3, the similarity between the physical and virtual environment is much lower, making alignment-based RDW more challenging [53]. This could be the reason why APF-Optimized controller performed as well as ARC, since ARC has the most rough alignment measure compared to Vis.-Poly. and Alignment-Optimized controllers. While most controllers had a worse performance in Environment 3 than in Environment 2, APF-RDW and APF-S2G actually performed better in Environment 3. The VE of Environment 3 has more empty space than that of Environment 2, resulting in more turns in the virtual paths and thus reducing the chance of walking straight towards obstacles after reset.

In Environment 4, there is a relatively large empty space in the PE while there are also large empty spaces in the VE, which is a key performance factor for alignment-based controllers and could be the reason why there was no significant performance difference between Alignment+APF-Optimized and Alignment-Optimized, since alignment measure was more dominant on performance than APF. Furthermore, as discussed previously, narrow areas have a huge impact on the performance of APF-S2G in terms of the number of resets, with this phenomenon more pronounced in Environment 4 due to the higher number of "dead ends" in the PE, which makes it difficult to "escape" from such a cramped space once the user entered it.

Limitations The area of the overlapping area between the physical and the virtual cells is used to measure $Alignment(\theta)$; however, considering area alone fails to distinguish cases that have the same size of overlapping area but with varying shapes that influence the RDW performance differently. In the experiment, the controllers were evaluated based on simulations with random paths in the form of piecewise Bézier curves. Simulation-based evaluation can be effective for quickly collecting data and understanding the controllers' strengths and weaknesses, but it usually does not take into account the user's perception and behavior evoked by the redirected steering, resulting in a loss of reality to a certain extent [12]. Since the piecewise Bézier segments can form a virtual path that visually resembles the global shape of the real user path, and the number of resets that Alignment-Optimized and Alignment+APF-Optimized achieved is significantly lower compared to the existing controllers,

the performance trends revealed in the simulation-based evaluation are likely to be preserved for a live user study, as found in [3]. Nevertheless, a live user study should be conducted in the future to evaluate the performance and understand the problems of controllers when applied in real applications. Another limitation is that, based on the proposed controllers, they all rely on the layout geometry of physical environment. When applied in a physical space with no layout information available, we need to do scene reconstruction beforehand or in real time. However, reconstructing the layout geometry of a physical environment in real time is still challenging and computationally expensive.

7 CONCLUSION AND FUTURE WORK

In this paper, we presented an optimization-driven framework for RDW controllers and, based on that, developed three novel RDW controllers: Alignment-Optimized, APF-Optimized, and Alignment+APF-Optimized. The optimization-driven framework allows us to measure the shape similarity between the physical and virtual walkable areas and to define new types of alignment definitions. Extensive simulations demonstrated that the Alignment-Optimized and Alignment+APF-Optimized controllers outperformed existing state-of-the-art controllers in terms of the number of resets. Among the three proposed controllers, the Alignment+APF-Optimized and Alignment-Optimized controllers exhibited superior performance compared to the APF-Optimized controller. However, the hypothesis that the Alignment+APF-Optimized controller outperforms the Alignment-Optimized controller is partially supported. The results also revealed that the Alignment+APF-Optimized and Alignment-Optimized controllers provide a more uniform likelihood distribution across distances between resets, as compared to the other redirection controllers. This study confirmed that alignment-based strategies for RDW is promising, deserving further studies for even more effective redirection controllers.

Artificial potential field provides useful information about obstacles in the physical environment. Although our study only partially supports the hypothesis that the Alignment+APF-Optimized controller outperforms the Alignment-Optimized controller, incorporating APF information into the optimization-driven alignment strategy is still worthy of further study. Current alignment-based strategies, including ours, utilize only information about the workable areas visible to the user in both physical and virtual environments. To further enhance their performance, incorporating information beyond the walkable areas in order to steer the user towards future safer areas in an optimization-driven framework could be a topic for future work. Moreover, currently existing alignment definition mainly depend on shape similarity between physical and virtual walkable areas. Alignment definitions that are other than shape similarity are desirable and expected to possess different advantageous features.

ACKNOWLEDGMENTS

This work was supported by the Ministry of Science and Technology, R.O.C., under Grant No. MOST 111-2221-E-A49-128 and the US National Science Foundation under Grant 2007435.

REFERENCES

- [1] M. Azmandian, T. Grechkin, M. Bolas, and E. Suma. The redirected walking toolkit: a unified development platform for exploring large virtual environments. In *2016 IEEE 2nd Workshop on Everyday Virtual Reality (WEVR)*, pp. 9–14. IEEE, 2016. 2
- [2] M. Azmandian, T. Grechkin, M. T. Bolas, and E. A. Suma. Physical space requirements for redirected walking: How size and shape affect performance. In *ICAT-EGVE*, pp. 93–100, 2015. 6
- [3] M. Azmandian, R. Yahata, T. Grechkin, J. Thomas, and E. S. Rosenberg. Validating simulation-based evaluation of redirected walking systems. *IEEE Transactions on Visualization and Computer Graphics*, 28(5):2288–2298, 2022. 9
- [4] S. V. Babu, H.-C. Huang, R. J. Teather, and J.-H. Chuang. Comparing the fidelity of contemporary pointing with controller interactions on performance of personal space target selection. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 404–413. IEEE, 2022. 7
- [5] E. R. Bachmann, E. Hodgson, C. Hoffbauer, and J. Messinger. Multi-user redirected walking and resetting using artificial potential fields. *IEEE Transactions on Visualization and Computer Graphics*, 25(5):2022–2031, 2019. 1, 2
- [6] G. P. Bingham and C. C. Pagano. The necessity of a perception–action approach to definite distance perception: Monocular distance perception to guide reaching. *Journal of Experimental Psychology: Human Perception and Performance*, 24(1):145, 1998. 7
- [7] L.-P. Cheng, E. Ofek, C. Holz, and A. D. Wilson. VRoamer: generating on-the-fly VR experiences while walking inside large, unknown real-world building environments. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 359–366. IEEE, 2019. 2
- [8] T. Dong, X. Chen, Y. Song, W. Ying, and J. Fan. Dynamic artificial potential fields for multi-user redirected walking. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 146–154. IEEE, 2020. 2, 3
- [9] T. Dong, Y. Shen, T. Gao, and J. Fan. Dynamic density-based redirected walking towards multi-user virtual environments. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pp. 626–634. IEEE, 2021. 3
- [10] E. Ebrahimi, A. Robb, L. S. Hartman, C. C. Pagano, and S. V. Babu. Effects of anthropomorphic fidelity of self-avatars on reach boundary estimation in immersive virtual environments. In *Proceedings of the 15th ACM Symposium on Applied Perception*, pp. 1–8, 2018. 7
- [11] O. Grodzewich and O. Romanko. Normalization and other topics in multi-objective optimization. In *Proceedings of the Fields–MITACS Industrial Problems Workshop*, 2006. 5
- [12] C. Hirt, Y. Kompis, C. Holz, and A. Kunz. The chaotic behavior of redirection - revisiting simulations in redirected walking. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 2022. 9
- [13] E. Hodgson, E. Bachmann, and D. Waller. Redirected walking to explore virtual environments: Assessing the potential for spatial interference. *ACM Transactions on Applied Perception (TAP)*, 8(4):1–22, 2008. 2, 4
- [14] S.-B. Jeon, S.-U. Kwon, J.-Y. Hwang, Y.-H. Cho, H. Kim, J. Park, and I.-K. Lee. Dynamic optimal space partitioning for redirected walking in multi-user environment. *ACM Transactions on Graphics (TOG)*, 41(4):1–14, 2022. 3
- [15] M. Keller and T. Tchilinguirian. Obstacles awareness methods from occupancy map for free walking in vr. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 1012–1013. IEEE, 2019. 2
- [16] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, vol. 4, pp. 1942–1948. IEEE, 1995. 5
- [17] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In *Autonomous Robot Vehicles*, pp. 396–404. Springer, 1986. 1, 2
- [18] B. Krogh. A generalized potential field approach to obstacle avoidance control. In *Proc. SME Conf. on Robotics Research: The Next Five Years and Beyond*, Bethlehem, PA, 1984, pp. 11–22, 1984. 1, 2
- [19] S.-U. Kwon, S.-B. Jeon, J.-Y. Hwang, Y.-H. Cho, J. Park, and I.-K. Lee. Infinite virtual space exploration using space tiling and perceivable reset at fixed positions. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 758–767. IEEE, 2022. 2
- [20] D.-Y. Lee, Y.-H. Cho, and I.-K. Lee. Real-time optimal planning for redirected walking using deep q-learning. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 63–71. IEEE, 2019. 2
- [21] Y.-J. Li, F. Steinicke, and M. Wang. A comprehensive review of redirected walking techniques: Taxonomy, methods, and future directions. *Journal of Computer Science and Technology*, 37(3):561–583, 2022. 1, 2
- [22] Y.-J. Li, M. Wang, F. Steinicke, and Q. Zhao. Openrdw: A redirected walking library and benchmark with multi-user, learning-based functionalities and state-of-the-art algorithms. In *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 21–30. IEEE, 2021. 2
- [23] C.-E. Lin, T. Y. Cheng, and X. Ma. Architect: Building interactive virtual experiences from physical affordances by bringing human-in-the-loop. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2020. 2
- [24] H. Liu, Z. Wang, A. Mazumdar, and C. Mousas. Virtual reality game level layout design for real environment constraints. *Graphics and Visual Computing*, 4:200020, 2021. 2
- [25] F. Marini and B. Walczak. Particle swarm optimization (PSO). a tutorial. *Chemometrics and Intelligent Laboratory Systems*, 149:153–165, 2015. 5
- [26] S. Marwecki and P. Baudisch. Scenograph: Fitting real-walking vr experiences into various tracking volumes. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, pp. 511–520, 2018. 2
- [27] G. Mavrotas. Effective implementation of the ϵ -constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computation*, 213(2):455–465, 2009. 5
- [28] J. Messinger, E. Hodgson, and E. R. Bachmann. Effects of tracking area shape and size on artificial potential field redirected walking. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 72–80. IEEE, 2019. 1, 2, 5, 6, 9
- [29] P. E. Napieralski, B. M. Altenhoff, J. W. Bertrand, L. O. Long, S. V. Babu, C. C. Pagano, J. Kern, and T. A. Davis. Near-field distance perception in real and virtual environments using both verbal and action responses. *ACM Transactions on Applied Perception*, 8(3), 8 2011. doi: 10.1145/2010325.2010328 7
- [30] T. Nescher, Y.-Y. Huang, and A. Kunz. Planning redirection techniques for optimal free walking experience using model predictive control. In *2014 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 111–118. IEEE, 2014. 2
- [31] N. C. Nilsson, T. Peck, G. Bruder, E. Hodgson, S. Serafin, M. Whitton, F. Steinicke, and E. S. Rosenberg. 15 years of research on redirected walking in immersive virtual environments. *IEEE Computer Graphics and Applications*, 38(2):44–56, 2018. 1, 2
- [32] C. C. Pagano and G. P. Bingham. Comparing measures of monocular distance perception: Verbal and reaching errors are not correlated. *Journal of Experimental Psychology: Human Perception and Performance*, 24(4):1037, 1998. 7
- [33] C. C. Pagano, R. P. Grutzmacher, and J. C. Jenkins. Comparing verbal and reaching responses to visually perceived egocentric distances. *Ecological Psychology*, 13(3):197–226, 2001. 7
- [34] T. C. Peck, H. Fuchs, and M. C. Whitton. Improved redirection with distractors: A large-scale-real-walking locomotion interface and its effect on navigation in virtual environments. In *2010 IEEE Virtual Reality Conference (VR)*, pp. 35–38. IEEE, 2010. 5
- [35] S. Razzaque. *Redirected walking*. PhD thesis, The University of North Carolina at Chapel Hill, 2005. 1, 2
- [36] S. Razzaque, Z. Kohn, and M. C. Whitton. Redirected walking. In *Proceedings of Eurographics*, 2001. 1, 2
- [37] R. A. Ruddle and S. Lessels. The benefits of using a walking interface to navigate virtual environments. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 16(1):1–18, 2009. 1
- [38] R. A. Ruddle, E. Volkova, and H. H. Bühlhoff. Walking improves your cognitive map in environments that are large-scale and large in extent. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 18(2):1–20, 2011. 1
- [39] A. L. Simeone, E. Velloso, and H. Gellersen. Substitutional reality: Using the physical environment to design virtual reality experiences. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 3307–3316, 2015. 2
- [40] K. H. Sing and W. Xie. Garden: A mixed reality experience combining virtual reality and 3D reconstruction. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 180–183, 2016. 2
- [41] M. Sra, S. Garrido-Jurado, C. Schmandt, and P. Maes. Procedurally generated virtual reality from 3D reconstructed physical space. In *Proceedings*

- of the 22nd ACM Conference on Virtual Reality Software and Technology, pp. 191–200, 2016. 2
- [42] F. Steinicke, G. Bruder, J. Jerald, H. Frenz, and M. Lappe. Estimation of detection thresholds for redirected walking techniques. *IEEE Transactions on Visualization and Computer Graphics*, 16(1):17–27, 2009. 1, 2, 4, 5
 - [43] F. Steinicke, G. Bruder, L. Kohli, J. Jerald, and K. Hinrichs. Taxonomy and implementation of redirection techniques for ubiquitous passive haptic feedback. In *2008 International Conference on Cyberworlds*, pp. 217–223. IEEE, 2008. 2
 - [44] R. R. Strauss, R. Ramanujan, A. Becker, and T. C. Peck. A steering algorithm for redirected walking using reinforcement learning. *IEEE Transactions on Visualization and Computer Graphics*, 26(5):1955–1963, 2020. 2
 - [45] E. A. Suma, S. Clark, D. Krum, S. Finkelstein, M. Bolas, and Z. Warte. Leveraging change blindness for redirection in virtual environments. In *2011 IEEE Virtual Reality Conference*, pp. 159–166. IEEE, 2011. 1
 - [46] S. Suri and J. O'Rourke. Worst-case optimal algorithms for constructing visibility polygons with holes. In *Proceedings of the second annual symposium on Computational geometry*, pp. 14–23, 1986. 3
 - [47] J. Thomas, C. Hutton Pospick, and E. Suma Rosenberg. Towards physically interactive virtual environments: Reactive alignment with redirected walking. In *26th ACM Symposium on Virtual Reality Software and Technology*, pp. 1–10, 2020. 2, 5
 - [48] J. Thomas and E. S. Rosenberg. A general reactive algorithm for redirected walking using artificial potential functions. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (IEEE VR)*, pp. 56–62. IEEE, 2019. 1, 2, 3, 4, 5, 6, 9
 - [49] M. Usuh, K. Arthur, M. C. Whitton, R. Bastos, A. Steed, M. Slater, and F. P. Brooks Jr. Walking> walking-in-place> flying, in virtual environments. In *Proceedings of the 26th Annual Conference on Computer graphics and Interactive Techniques*, pp. 359–364, 1999. 1
 - [50] I. Valentini, G. Ballestin, C. Bassano, F. Solari, and M. Chessa. Improving obstacle awareness to enhance interaction in virtual reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 44–52. IEEE, 2020. 2
 - [51] B. Williams, G. Narasimham, B. Rump, T. P. McNamara, T. H. Carr, J. Rieser, and B. Bodenheimer. Exploring large virtual environments with an hmd when physical space is limited. In *Proceedings of the 4th symposium on Applied perception in graphics and visualization*, pp. 41–48, 2007. 2, 5
 - [52] N. L. Williams, A. Bera, and D. Manocha. ARC: Alignment-based redirection controller for redirected walking in complex environments. *IEEE Transactions on Visualization and Computer Graphics*, 27(5):2535–2544, 2021. 2, 3, 5, 6
 - [53] N. L. Williams, A. Bera, and D. Manocha. Redirected walking in static and dynamic scenes using visibility polygons. *IEEE Transactions on Visualization and Computer Graphics*, 27(11):4267–4277, 2021. 2, 3, 5, 6, 9
 - [54] N. L. Williams, A. Bera, and D. Manocha. ENI: Quantifying environment compatibility for natural walking in virtual reality. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 419–427. IEEE, 2022. 3
 - [55] X. Xie, Q. Lin, H. Wu, G. Narasimham, T. P. McNamara, J. Rieser, and B. Bodenheimer. A system for exploring large virtual environments that combines scaled translational gain and interventions. In *Proceedings of the 7th Symposium on Applied Perception in Graphics and Visualization*, pp. 65–72, 2010. 2
 - [56] S.-Z. Xu, T.-Q. Liu, J.-H. Liu, S. Zollmann, and S.-H. Zhang. Making resets away from targets: POI aware redirected walking. *IEEE Transactions on Visualization and Computer Graphics*, 28(11):3778–3787, 2022. 2
 - [57] C. A. Zambaka, B. C. Lok, S. V. Babu, A. C. Ulinski, and L. F. Hodges. Comparison of path visualizations and cognitive measures relative to travel technique in a virtual environment. *IEEE Transactions on Visualization and Computer Graphics*, 11(6):694–705, 2005. 1
 - [58] S.-H. Zhang, C.-H. Chen, and S. Zollmann. One-step out-of-place resetting for redirected walking in vr. *IEEE Transactions on Visualization and Computer Graphics*, 2022. 2
 - [59] M. A. Zmuda, J. L. Wonser, E. R. Bachmann, and E. Hodgson. Optimizing constrained-environment redirected walking instructions using search techniques. *IEEE Transactions on Visualization and Computer Graphics*, 19(11):1872–1884, 2013. 2